

Course: Architecture and Programming Principles

Instructor: Professor Shaikh

Student Name: Yaoting Jiang

Date: Dec 1st, 2021

PAYMENT METHOD RESOURCES

Use Case: Get payment method

- **Resource(s):** payment method
- **Request URL:** http://localhost:5000/payment_method
- **Request body:** N/A
- **Response:**

The screenshot shows a REST client interface with the following details:

- Request:** GET http://localhost:5000/payment_method
- Status:** 200 OK, 27 ms, 344 B
- Response Body (JSON):**

```
{  "id": "61a7df80b3d272ee25081797",  "credit_card": "2000 0000 0000 0000",  "expiration_date": "11/11/21",  "security_code": "000",  "billing_address": "111 toyama dr, Sunnyvale, CA, 94089"}
```

Use Case: Create a payment method

- **Resource(s):** payment method
- **Request URL:** `http://localhost:5000/payment_method?credit_card=4400 8888 9999 0000&expiration_date=11/11/21&security_code=111&billing_address=160 Pleasant Dr, Boston, MA, 02148`
- **Request body:** N/A
- **Response:**

The screenshot displays a REST client interface with a POST request configured and a successful JSON response.

Request Configuration:

- Method:** POST
- URL:** `http://localhost:5000/payment_method?credit_card=4400 8888 9999 0000&expiration_date=11/11/21&security_code=111&billing_address=160 Pleasant Dr, Boston, MA, 02148`
- Params:** A table of query parameters is shown below.

Key	Value	Description
<input checked="" type="checkbox"/> credit_card	4400 8888 9999 0000	
<input checked="" type="checkbox"/> expiration_date	11/11/21	
<input checked="" type="checkbox"/> security_code	111	
<input checked="" type="checkbox"/> billing_address	160 Pleasant Dr, Boston, MA, 02148	

Response:

Status: 200 OK, 8 ms, 341 B

Body (JSON):

```
1 {
2   "_id": {
3     "$oid": "61a7e0b88275062967d65f6a"
4   },
5   "credit_card": "4400 8888 9999 0000",
6   "expiration_date": "11/11/21",
7   "security_code": "111",
8   "billing_address": "160 Pleasant Dr, Boston, MA, 02148"
9 }
```

Use Case: Update a payment method

- **Resource(s):** payment method
- **Request URL:** `http://localhost:5000/payment_method/61a7df80b3d272ee25081797?credit_card=3300 0000 1111 2222&expiration_date=10/11/21&security_code=222&billing_address=222 toyama dr, Sunnyvale, CA 94089`
- **Request body:** shown in the screenshot below
- **Response:**

The screenshot displays a REST client interface with a PATCH request to `http://localhost:5000/payment_method/61a7df80b3d272ee25081797?credit_card=3300 0000 1111 2222&expiration_date=10/11/21&security_code=222&billing_address=222 toyama dr, Sunnyvale, CA 94089`. The request parameters are listed in a table below.

Key	Value	Description
<input checked="" type="checkbox"/> credit_card	3300 0000 1111 2222	
<input checked="" type="checkbox"/> expiration_date	10/11/21	
<input checked="" type="checkbox"/> security_code	222	
<input checked="" type="checkbox"/> billing_address	222 toyama dr, Sunnyvale, CA 94089	

The response is a JSON object, displayed in the 'Body' tab:

```
1 {
2   "_id": {
3     "$oid": "61a7df80b3d272ee25081797"
4   },
5   "credit_card": "1100 0000 3333 0000",
6   "expiration_date": "10/10/25",
7   "security_code": "000",
8   "billing_address": "360 Huntington Ave, Boston, MA, 02148"
9 }
```

At the bottom right, there are icons for 'Runner', 'Trash', and a help icon.

Use Case: Delete a payment method

- **Resource(s):** payment method
- **Request URL:** http://localhost:5000/payment_method/61a7df80b3d272ee25081797
- **Request body:** N/A
- **Response:**

GET http://localhost:50...

DEL http://localhost:50...

+ ...

No Environment

http://localhost:5000/payment_method/61a7df80b3d272ee25081797

Save

DELETE

http://localhost:5000/payment_method/61a7df80b3d272ee25081797

Send

Params

Auth

Headers (8)

Body

Pre-req.

Tests

Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body

Cookies

Headers (4)

Test Results

200 OK

18 ms

146 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

1

RATING RESOURCES

Use Case: Get rating

- **Resource(s):** rating
- **Request URL:** http://localhost:5000/rating
- **Request body:** N/A
- **Response:**

GET http://localhost:50... GET http://localhost:50... + ...

No Environment

http://localhost:5000/rating

Save

Send

GET

http://localhost:5000/rating

Send

Params

Auth

Headers (8)

Body

Pre-req.

Tests

Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body

Cookies

Headers (4)

Test Results

200 OK

13 ms

236 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   {
3     "_id": {
4       "$oid": "61a971ce3a5e8c45587d4b55"
5     },
6     "score": 5,
7     "dog_walker": "TEST DOGWALKER"
8   }
9 }
```

Use Case: create a rating

- **Resource(s):** rating
- **Request URL:** `http://localhost:5000/rating?score=4&dog_walker=Test Dogwalker2`
- **Request body:** N/A
- **Response:**

The screenshot shows a REST client interface with a POST request to `http://localhost:5000/rating?score=4&dog_walker=Test Dogwalker2`. The response is a 200 OK status with a 14 ms latency and 235 B of data. The response body is displayed in JSON format, showing a successful rating creation with a unique ID and the provided score and dog walker name.

Request Details:

- Method: POST
- URL: `http://localhost:5000/rating?score=4&dog_walker=Test Dogwalker2`
- Params: score=4, dog_walker=Test Dogwalker2

Response Details:

- Status: 200 OK
- Latency: 14 ms
- Size: 235 B

Response Body (JSON):

```
{
  "_id": {
    "$oid": "61a974433a5e8c45587d4b56"
  },
  "score": 4,
  "dog_walker": "Test Dogwalker2"
}
```

Use Case: update a rating

- **Resource(s):** rating
- **Request URL:** `http://localhost:5000/rating/61a971ce3a5e8c45587d4b55`
- **Request body:** as shown in the screenshot below

Response:

The screenshot displays a REST client interface with a PATCH request configured for the URL `http://localhost:5000/rating/61a971ce3a5e8c45587d4b55`. The request body is a JSON object with the following structure:

```
1 {
2   "score": 3,
3   "dog_walker": "Test Dogwalker2"
4 }
```

The response section shows a 200 OK status with a response time of 26 ms and a body size of 235 B. The response body is displayed in a pretty-printed JSON format:

```
1 {
2   "_id": {
3     "$oid": "61a971ce3a5e8c45587d4b55"
4   },
5   "score": 3,
6   "dog_walker": "Test Dogwalker2"
7 }
```

Use Case: delete a rating

- **Resource(s):** rating
- **Request URL:** `http://localhost:5000/rating?score=4&dog_walker=Test Dogwalker2`
- **Request body:** as shown in the screenshot below

Response:

The screenshot displays a REST client interface with the following components:

- Top Bar:** Shows the current request method as `DEL` and the URL `http://localhost:5000/rating/61a971ce3a5e8c45587d4b55`. It also includes a "No Environment" dropdown and a "Save" button.
- Request Configuration:** A dropdown menu shows `DELETE` as the method. The "Send" button is visible.
- Request Body:** The "Body" tab is selected, showing a JSON payload:

```
1 {
2   "score": 3,
3   "dog_walker": "Test Dogwalker2"
4 }
```
- Response Section:** The "Body" tab is selected, showing the response status `200 OK`, a response time of `66 ms`, and a response size of `146 B`. The response body is currently empty, showing only the opening bracket of an array `[]`.

I'm taking a grace period for this assignment

49-787 Request for Grace Period - Pass 3

yaotingj@andrew.cmu.edu [Switch account](#)

 Draft saved

Your email will be recorded when you submit this form

* Required

Use this form to redeem a pass for a 48-hour extension on a valid assignment per course syllabus. (Note: Requests for invalid assignments will not be honored, though your pass will be marked as used!)

Title of assignment for which grace period is requested *

Project 7B

Reason for requesting grace period (Do not put personal/private information) *

My ~~mongodb~~ is not working again for some re

A copy of your responses will be emailed to yaotingj@andrew.cmu.edu.

Submit

[Clear form](#)

Never submit passwords through Google Forms.

This form was created inside of Carnegie Mellon University Silicon Valley campus. [Report Abuse](#)